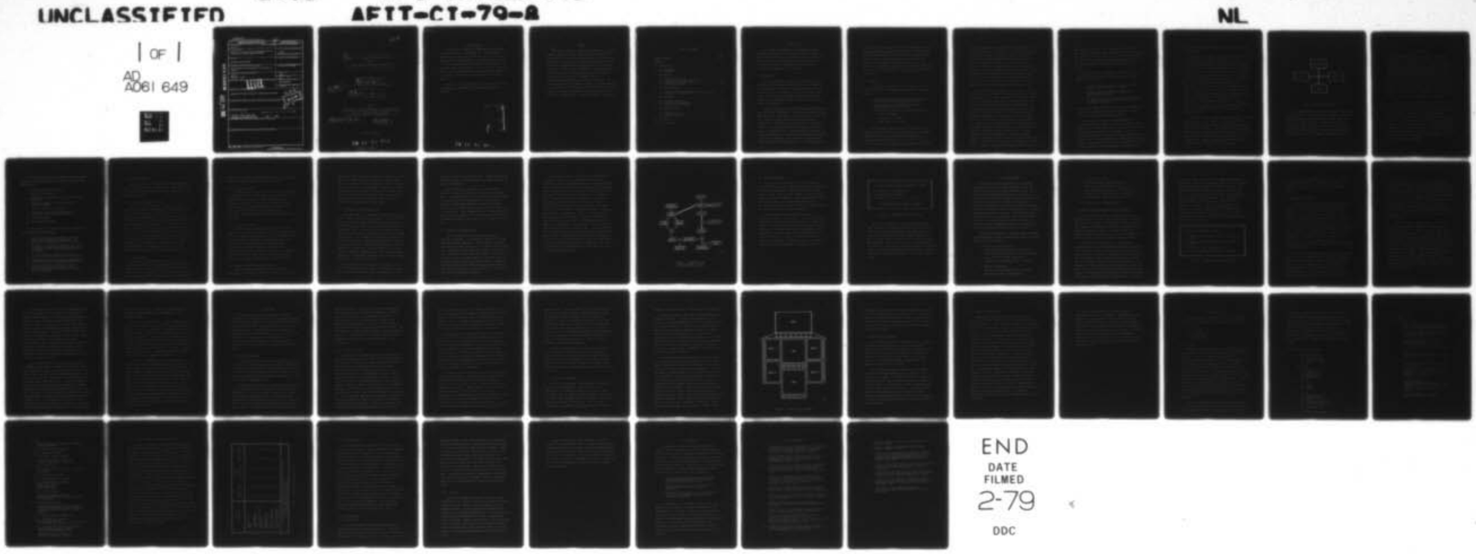
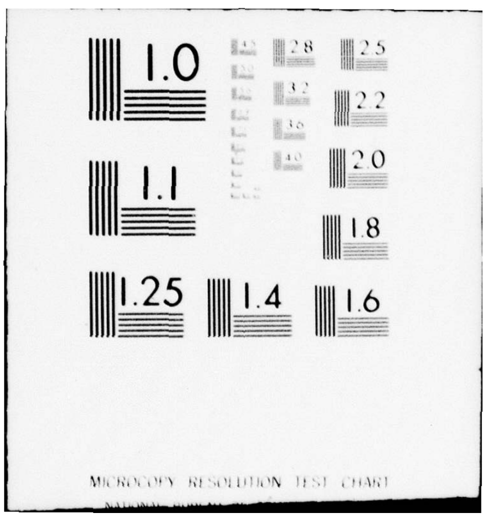
AD-A061 649 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO F/G 1/2
ANALYSIS OF THE FUNCTIONAL REQUIREMENTS FOR AN INTELLIGENT AIRB--ETC(U)
1978 R I MORISHIGE
UNCLASSIFIED AFIT-CI-79-8 NL

| OF |

AD
A061 649



END
DATE
FILMED
2-79
DDC


1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6
MICROCOPY RESOLUTION TEST CHART

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER CI 79-8	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) Analysis of the Functional Requirements for an Intelligent Airborne Computer System	5. TYPE OF REPORT & PERIOD COVERED Thesis
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Ronald Itsuo Morishige	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS AFIT Studn-t at the University of Illinois at Urbana	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS AFIT/CI WPAFB OH 45433	12. REPORT DATE 1978
	13. NUMBER OF PAGES 44
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report) UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

LEVEL

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release, Distribution Unlimited

DDC
RECEIVED
NOV 29 1978
F

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

APPROVED FOR PUBLIC RELEASE AFR 190-17.

JOSEPH P. HIPPS, Major, USAF
Director of Information, AFIT

NOV 9 1978

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

79-8

(6) ANALYSIS OF THE FUNCTIONAL REQUIREMENTS

FOR AN

INTELLIGENT AIRBORNE COMPUTER SYSTEM.

BY

(10) RONALD ITSUO MORISHIGE

B.S., United States Air Force Academy, 1967

(12) 49p.

(9) Master's thesis,

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 1978

(14) AFIT-CI-79-8

(11) 1978

Urbana, Illinois

ACKNOWLEDGEMENT

I would like to express my appreciation to the George Olmsted Foundation of Washington, D.C. for scholarship benefits and to the United States Air Force for providing an opportunity for graduate study. I wish to thank Dr. Robert T. Chien, my adviser, for his patience, encouragement, and guidance during this research. To Paul Davis, for taking the time to get me interested in the area, and to David Chen, for his technical advice and good conversation, my sincerest thanks.

Finally, my love and gratitude to Jerry, Mark, and Greg, for enduring my inattention and short temper during my studies.

PREFACE

The use of artificial intelligence methods in an airborne computer system can enhance flight safety by reducing the possibility of pilot error caused by inadequate or misleading information. The intelligent computer system would have the ability to screen information for relevancy to the current situation. The selection of information requires knowledge of the context which may depend upon the phase of flight, the condition of the aircraft, or external factors. The phase-based context might be represented by scripts. The condition-based and the external context can be represented by a cause-effect net. Knowledge of context will not only enable priority resolution of information, but also definition of goals for generating plans to cope with abnormal situations.

Table of Contents

1. INTRODUCTION

The essential attribute of an intelligent airborne computer system will be its ability to provide information which will help the pilot "stay ahead of the aircraft". The better the pilot can anticipate the future state of the aircraft, the better he can make good, safe decisions concerning the operation of the aircraft.

1.1 Motivation

This paper presents an analysis of the functions of an advanced airborne computer system which utilizes artificial intelligence (AI) techniques. The airborne application was selected because it represents a very specific and limited domain, yet is still rich enough to provide a meaningful application. Furthermore, there is potential for immediate practical application.

The safe operation of a modern aircraft depends primarily upon the pilot's performance. It is believed that the pilot's performance can be improved and flight safety can be enhanced by an advanced computer system that provides relevant information to the pilot on a real-time basis. The primary objective is to prevent errors that are caused by the pilot taking an improper course of action because of inadequate or misleading information. As aircraft become more reliable, the corresponding experience level of pilots goes down because fewer experience actual difficulties. It is not uncommon for

pilots with several thousand hours of jet flying time to have never experienced an actual engine failure. At the same time the increasing complexity of aircraft further aggravates the problem. Hence, it would be advantageous to utilize the computer system knowledge base to supplement the pilot's experience. A reduction in peak workload would be beneficial because accidents are frequently associated with high workload situations.

1.2 Criteria

A survey by the Lockeed California Company [7] of 500 pilots resulted in the following *criteria for an information system*.

1. Information should help keep the pilot thinking ahead of the current state of the aircraft.
2. Visual displays are preferred for receiving information and color is beneficial.
3. Displays should answer the following questions:
    1. "How am I doing?"
    2. "How well am I doing?"
    3. "What should I be doing?"

To be compatible with the pilot, the computer system should not increase the workload of the pilot. The computer system should act like an additional, highly-trained crewmember whose primary task is to assist the pilot. The value of any kind of assistance is diminished if it increases

the pilot's workload by requiring excessive pilot inputs or displaying extraneous information. A conventional approach to the airborne computer system might attempt to provide the pilot with all related information immediately and let the pilot decide what is relevant. This approach causes an increase in the pilot's workload because he must examine all of the data that the computer system presents to him. Earlier information systems like the Hughes Master Monitor Display [10] and the Boeing Integrated Information Presentation and Control System (IIPACS) [1] did not attempt to analyze the raw data for relevancy and suffered from an information glut.

The AI approach to the problem is to use the computer to screen the data and present the information which is relevant to the situation in order of priority. The screening implies a priority structure which must be sensitive to the general state or context of the aircraft and its environment. In addition, the intelligent onboard computer should ease the problem solving task of the pilot by providing necessary procedural information. The computer system can further reduce the pilot's workload by executing tasks delegated to it. Other tasks could include automatic reaction to certain emergency conditions. Earlier work in this area resulted in a computer aided decision making system (CADM) [3] which detected and corrected failures in electrical and fuel systems. Detection was done by using data driven programs or DEMONS. A DEMON is a program that is activated when specific data is received. The data could be sensor values, time, or

any other type of data. Error correction procedures were generalized and operated upon a semantic net representation of the aircraft. The CADM system used a fixed strategy which was not sensitive to changes in the flight environment. This lack of context sensitivity reduced the effectiveness of the system.

The Digital Avionics Information System (DAIS) [9] represents a significant step toward an advanced computer system. The DAIS design

1. recognizes the importance of a fully integrated computer system in advanced aircraft,
2. attempts to tailor computer operation to different phases of flight by selecting the appropriate Master Mode,
3. and attempts to control pilot workload and treat the pilot as a manager rather than as a slave to the computer system.

However, DAIS depends upon the pilot to change the Master Mode and to make inputs which might not be necessary if more complete context information were available.

This paper will review two promising methods for representing different types of context information. There seem to be three types of context. One type of context is the phase-based context which is of a temporal nature. It reflects how and when systems change during different phases of the flight. A promising method of representing phase-based context is the script concept of Schank and Abelson [16]. The scripts provide the normal background information for each

phase, as well as a mechanism for anticipating future states of the aircraft.

Another type of context is condition-based context. It is largely aircraft and system dependent, and it does not change for a given aircraft. This would include information on such items as fuel system management, hydraulic redundancy and backup systems, the number of generators, and sequencing for the operation of the landing gear. The condition-based context would probably be best represented by a cause-effect net. The third type of context is the external or environmental context. This is largely aircraft independent and may change at any time. Some factors considered in the external context include weather, aircraft location, terrain, darkness, conflicting traffic, location of alternate airfields, and similar types of information. A potential method for representing the latter two types of context might utilize a cause-effect net similar to the Common-Sense Algorithm [13, 14].

The basic components which an intelligent computer system should possess include a monitor, a plan generator, an executive, and a knowlege base. The knowledge base would be common to all components. Control would be semi-autonomous with the executive providing the high level directives and the monitor and plan generator capable of performing its function independently. Figure 1 depicts the arrangement of the components with a common bus for communications.

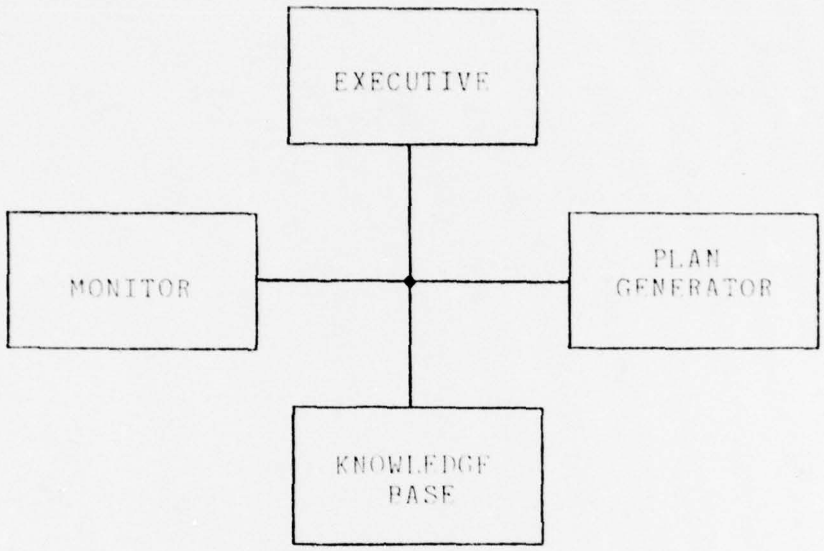


Figure 1 General Organization

It is apparent that the intelligent computer system must be highly reliable since false alarms or undetected deviations could be disasterous or at least significantly lessen the value of the computer system. It must utilize extensive self-testing, error correction, redundancy, and be able to operate in a degraded mode. The system must be designed to fail gradually rather than suddenly and without warning.

2. THE MONITOR

The primary purpose of the monitor is to provide continuous, real-time information about the state of the aircraft and its systems. The use of a computer monitor can significantly reduce pilot workload by relieving him of the tedious task of monitoring numerous instrument readings and mentally integrating the raw data to determine the overall systems performance. For example, engine performance is now monitored by checking and correlating the indications for EGT, RPM, fuel flow, nozzle position, EPR, etc. Normally, it would be much easier for the pilot to monitor a single thrust reading. The complete set of parameters would be available to the pilot on request or automatically when needed.

It is generally acknowledged that certain phases of flight are characterized by a high workload for the pilot and crew. During high workload periods, the quality and quantity of the normal monitoring is likely to be diminished. Deviations are more likely to go undetected for longer periods of time. An unnoticed deviation can frequently compound the hazard of a high workload situation. The monitor would not be affected by the increased workload since it will be designed to have excess capacity over peak load situations. Nor would the monitor be affected by typical human failings such as boredom, fatigue, or fixation.

The intelligent monitor would require several supporting functions. The capabilities for the monitor should include the following:

1. Verification of sensor output.

2. Detection of deviations.

3. Diagnosis of a common cause for related sensor indications.

4. Reporting diagnosis to the executive for priority resolution.

5. Automatic checklist presentation.

6. Generation of a history of deviations and significant events for later analysis.

7. Real-time response

8. Degraded mode operation

The basic sequence of operations for the monitor should probably follow the steps below.

1. Sensors provide direct information about the state of the aircraft and its systems. All significant measurable parameters are sensed.

2. The sensor readings are verified for logical consistency. Any sensor indication which is not consistent is given a lower plausibility and is so reported.

3. The verified sensor readings are compared against a context sensitive model for agreement.

4. A diagnosis is made based upon the deviations to determine if there is a probable common cause.

5. Any diagnosis and the supporting evidence (sensor indications) are reported to the executive for relay to the pilot and recorded for later analysis.

2.1 Sensor Verification

Sensor verification is the process of establishing the logical consistency among related sensors. The nature of the consistency will depend upon the physical constraints of the sensor being checked.

2.1.1 Exhaust gas temperature (EGT)

This is a primary indication of thrust. A higher EGT will normally indicate a higher thrust. Related sensors are fuel flow, RPM, EPR (ratio of turbine pressure to inlet pressure), air temperature, etc. Normally the fuel control schedules fuel to the engine based upon throttle position, air temperature, RPM, EGT, and compressor discharge pressure. Proper operation of the fuel control depends upon correct sensor data. High EGT is normally associated with high fuel flow, high EPR, and usually high RPM. If the EGT trend were to follow the other parameters, then the EGT sensor would probably be operating properly. If a change in EGT were reported without a corresponding change in the related parameters, then the EGT sensor would probably be in error.

2.1.2 Fuel quantity

The quantity of fuel within a particular tank can be checked by transferring known quantities of fuel into or out of the tank being checked. Also, because the flow rate is usually known, it is possible to integrate to fuel flow over a period of time to derive a value by which the quantity should

change. The indicated quantity of fuel would also be compared against the planned quantity from the flight plan.

2.1.3 Hydraulic Pressure

The hydraulic pressure is a measure of the performance of the hydraulic pumps. Related sensors include hydraulic fluid quantity, hydraulic pump rpm, fluid temperature, and accumulator pressure. Sensors which directly report the resultant action of hydraulic pressure are also necessary. These would include flight control surface position indicators, other hydraulically actuated mechanism position indicators.

2.1.4 Inertial Platform

The verification of this type of sensor is a problem because it is a reference standard. While direct verification of an accelerometer can only be done by using redundancy, the resulting position information of the inertial navigation system can be directly checked by independent means. Similarly, the attitude information can be verified by checking the aircraft's flight characteristics. For example, if the aircraft is not wings level, it will tend to turn. Pitch can be checked against angle of attack and airspeed.

2.1.5 Angle of Attack/Pitot-static inputs

The relationship between the angle of attack and pitot-static data varies as a function of the forces acting

upon the aircraft: lift, drag, weight, and thrust. Lift is a function of angle of attack, airspeed, and configuration. Drag also depends on the same parameters. Thrust is simply the effective thrust of the engines. Weight is the effective weight of the aircraft (takes G-loading into account). The angle of attack and the pitot-static data can be correlated with additional data about the engine thrust, aircraft acceleration, attitude, and aircraft weight.

2.2 Determination of the Normal State

The normal state of a system may depend upon the context of the aircraft. The phase-based context is often reflected in the crew checklist where different modes are specificed for different phases of flight. For example, during landing and takeoff the position of the landing gear should be down, but during cruise, the position should be up and locked. Besides configuaration, other examples of systems where parameters are sensitive to the phase of flight are pressurization, the fuel system, and navigation. The intelligent computer system can determine the normal state of some parameters by the use of scripts. Flight plan information will also provide reference information on planned fuel quantity, route of flight, cruise speeds, etc. to monitor navigation.

Often the normal state or range of a system will vary with the condition of the aircraft. For example, if there has been a generator malfunction, then the normal state for the

generator bus tie relay should be open. Similarly during the transfer of fuel, the affected valves and pumps should be open and on, respectively.

Factors which are external to the aircraft such as weather or terrain may also have an affect upon the normal state. Bad weather often calls for increased safety margins. In gusty winds or suspected wind-shear, it is normal to increase airspeed when maneuvering close to the ground. Braking on a wet runway will not provide the same deceleration as a dry runway. A wet runway will have a considerable affect upon V_1 (critical engine failure speed), which may affect the recommended procedures during the loss of an engine on takeoff roll.

2.3 Diagnosis of Probable Cause

Often a malfunction will cause several sensors to indicate a deviation. It is therefore essential to isolate the cause from the symptoms. A simple example of diagnosis with multiple sensor indications is an engine flameout caused by a malfunctioning fuel boost pump. Within a short span of time, the monitor should detect fuel pressure low, RPM low, EGT low, EPR low, associated generator output off, and low hydraulic pressure on the associated system. A conventional system would provide the pilot with every sensor indication and depend upon the pilot and his experience to analyze the data and draw the proper conclusion.

The following method for diagnosis uses the cause-effect net shown in figure 2. The cause-effect net would be represented in the computer with directional links between the nodes. Failure of the number 2 boost pump with the cross-feed valve closed, will cause fuel pressure to drop, which in turn will retard fuel flow, which in turn will cause combustion to cease. Combustion requires the simultaneous and continuous existence of not only fuel flow, but also air flow and ignition. Without combustion, the engine will stop running, which will cause the number 2 generator and number 2 hydraulic pump to fail. If the number 2 AC bus is hot because it is receiving power from the number 1 generator, and the boost pump switch is on, then *there is an inconsistency across the* number 2 boost pump node. That is, there is no fuel pressure, even though the pump has electrical power. This inconsistency can be found from any point in the net that is affected by the inconsistency by backing up the links until the inconsistency is found. The result of the diagnosis would be presented with a list of supporting indications or consequences. This type of information will particularly help less experienced pilots cope with unusual situations.

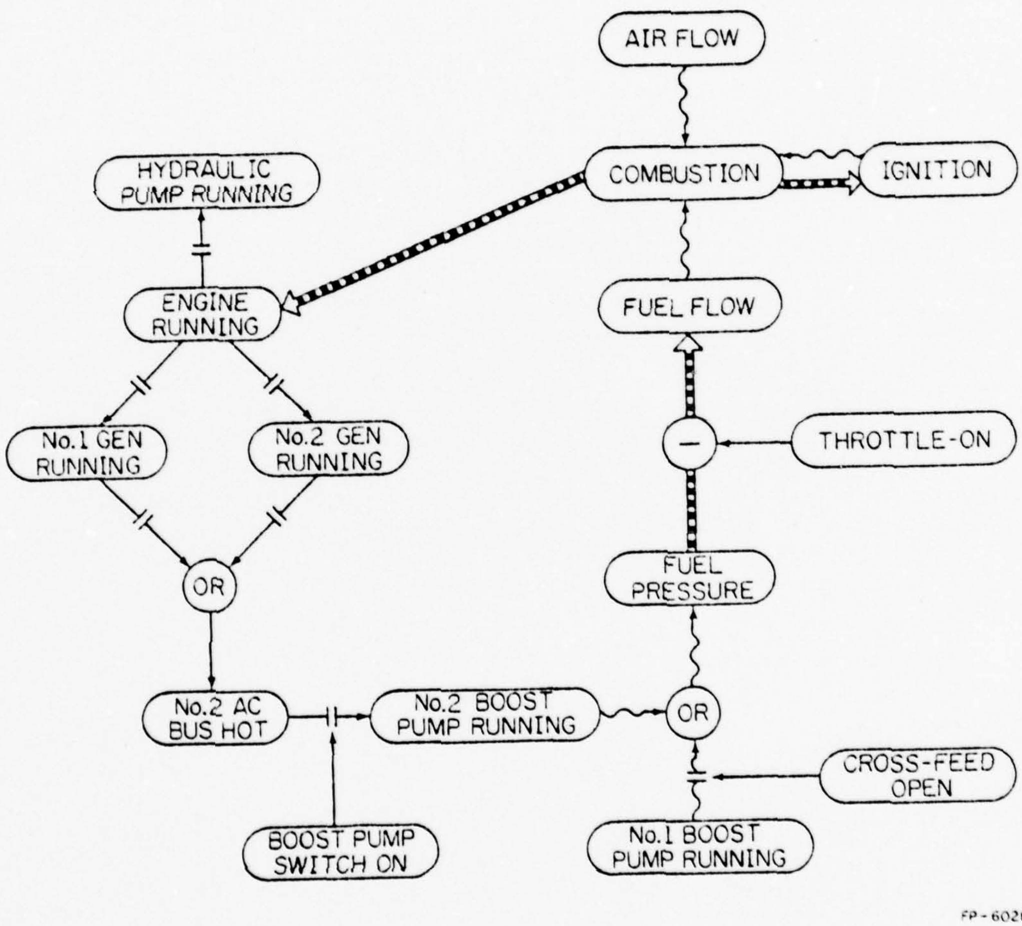


Figure 2 Cause-Effect Net
Based on Rieger [13,14]
For Basic Engine Operation

2.4 Automatic Checklist

The purpose of the checklist is to insure critical actions are not forgotten. Human limitations necessitate that the checklist procedures be brief and cover only the most important items. Therefore, the omission of any item could result in a hazardous condition. There are two basic types of checklist actions. One is monitoring and the other is preparation. The former is well handled by the basic monitor. The latter implies more planning information.

The preparation for an anticipated maneuver (e.g. landing) may require several actions. The script can provide information on what actions need to be taken to prepare for a maneuver. The preparations include aircraft preparation such as lowering the landing gear, but more importantly, pilot preparation. The pilot would be positively notified that a specific checklist for a portion of the flight had been successfully completed. For example, prior to the approach to landing the display might appear as depicted in figure 3.

APPROACH TO FIELD CHECKLIST COMPLETED - NORMAL

1. ILS RWY 25L to Los Angeles Intl
2. APPROACH SPEED = 140 KTS
3. MINIMUMS: (200 - 1/2)   DH = 326' MSL

VERIFY APPROACH AND LANDING CLEARANCE

Figure 3   Simulated Checklist Display

This kind of information, if presented at the proper time, can help to keep the pilot ahead of the aircraft. In order to provide this information at the proper time, it is necessary to have flight plan and clearance information as well as flight scripts to provide timing and synchronization for the automatic checklist. It should be emphasized that the information in figure 3 merely supports the Vertical Situation Display (or the Heads-Up Display) and the Horizontal Situation Display which are the primary sources of information during flight.

3. THE PLAN GENERATOR

The objective of the plan generator is to provide procedural information to permit the pilot to cope with abnormal situations in the safest possible manner. This is accomplished by providing the necessary information on a timely basis. The normal sequence of operation for the plan generator would start with a signal from the monitor indicating an abnormal condition and a diagnosis. The plan generator would then provide a canned plan from its standard repertoire or generate a new plan. The primary reason for canned plans is to assure minimum reaction time to abnormal conditions. A canned plan might result in the use of automatic severity abatement procedures to further reduce reaction time under certain well defined situations.

The general guidelines for operation during an abnormal situation applies to every situation in any type of aircraft. These rules are as follows:

1. Maintain aircraft control.
   This means that the priority of the problem should not override the basic "safety of flight" rules. Aircraft control should not be sacraficed in an attempt to solve a problem.

2. Analyze the situation.
   The pilot and the monitor must collectively determine the cause of the problem.

3. Take the proper action.
   The plan generator should come up with a sequence of actions that will correct, or at least ameliorate the situation. Failing that, it might try to generate a plan to cope with the situation until a safe landing can be made.

4. Land as soon as conditions permit.

3.1 Automatic Severity Abatement Procedures (ASAP)

Certain situations become progressively worse until corrective action is taken. The purpose of ASAP is to utilize the speed of the computer to limit damage to the aircraft by rapid response to time-critical situations by automatically initiating corrective action. For safety reasons, the computer would not initiate any action that was irreversible. This would enable the pilot to override any ASAP action.

In the case of an engine fire, it is desirable to minimize damage to the aircraft without increasing the hazard to the crew and passengers. Shutting down the engine would minimize damage to the engine, but before doing so, the system must verify that the thrust remaining after engine shutdown will be sufficient to maintain a safe flight condition. For example, if a three engine aircraft experiences an engine fire while flying at 10,000' MSL and the remaining two engines were capable of maintaining the current flight condition, then ASAP would automatically shutdown the affected engine. In another

case, a two engine aircraft on final approach experiences an engine fire. Here, if the remaining thrust on the good engine were not adequate to maintain a safe flight condition, the engine would not be automatically shutdown by ASAP because doing so could increase the actual hazard. The determination of a safe flight condition requires a rapid check of the thrust available, local terrain, and current flight parameters. Generally, caution is necessary when at low altitude, low airspeed, and with high drag configurations associated with landing. The pilot would still receive a warning about the fire and the applicable emergency procedure text as shown in figure 4.

ENGINE FIRE (EMERGENCY PROCEDURE)

1. Throttle (NO. 2 engine) - RETARD (if conditions permit)

2. (If fire continues) Throttle (NO. 2 engine) - CUTOFF

3. Fire Handle (NO. 2 engine) - PULL

4. Fire extinguisher (NO. 2 engine) - DISPENSE

Figure 4 Engine Fire Procedure Display

In the above emergency procedure, only the first two steps would be ASAP capable. The last three steps are examples of irreversible procedures and hence never ASAP capable.

3.2 Generation of New Plans

It would be unreasonable to assume that contingency plans for all possible deviations could be anticipated, much less programmed. For this reason the computer system should possess the capability for generating new plans. Plan generators, planners, or problem-solvers might be categorized into two broad types. There are the broad problem solvers which seek to find a general methodology applicable to a broad class of problems. This approach has met with rather limited success, which is not surprising when the nature of the task is examined. An alternate approach using specialized knowledge bases has had more demonstrable results. Some leading examples of specialist systems are MYCIN [17], DENDRAL [2], and MACSYMA [12].

The *ad hoc* approach would seem most useful at the present time because the aircraft and the flight domain provide some natural contraints which can simplify the plan generation task. The goal is always the same - to land safely at a suitable airfield. A flight plan and the flight script provide short-range goals for the operation of the aircraft. The possible actions which can be taken are limited to a

fixed, though large set which can be controlled, either directly or indirectly, from the cockpit. Many actions, such as repairs, are precluded because of inaccessibility of system components during flight. Hence, redundany is commonly used in the design of aircraft systems which simplifies the problem solving task. Another constraint is that many controls are binary - either a system or function is ON or OFF. In addition, most of the interactions among the systems of the aircraft have been studied in extreme detail and are well understood. These considerations will facilitate the ad hoc plan generator.

An earlier airborne computer system, CADM [3], which involved the generation of actions to correct abnormal situations was limited in certain respects. While CADM was able to solve a large class of simulated malfunctions, it was not sensitive to changes in context during a normal flight. The CADM error correction procedure was developed by the use of a fixed strategy. CADM assumed that the pilot was always right and did not interact constructively with the pilot during the planning stage. Hence, CADM could not utilize the pilot's knowledge of the aircraft and its systems, of procedures, or of high level goals. Finally, if CADM failed, it did not provide the pilot adequate information about the reasons for failure.

The presence of the pilot is also a consideration that must be incorporated into the design of an intelligent airborne computer system. The pilot, with his experience and training, can be a valuable source of information during periods of abnormal operation. For example, the problem of computer planning in incompletely specified domains [4] has not really been solved. The pilot can provide high level guidance to minimize the possibility of the computer system becoming side-tracked. Pilot inputs could be the specification of goals, resolution of priority, resolution of conflicting information, or the specification of current limitations. While excessive pilot dependence is not desirable, some interaction would be of mutual value to the pilot and the computer system.

The plan generator need not be able to solve every problem in minute detail. It should do an excellent job on most situations, a reasonable job on most of the other jobs, and fail only partially on the remaining few problems. The concept of the use of a partial plan [5] emphasizes the need for sufficiency rather than detail. It must be emphasized that a partial plan which may not be adequate for computer implementation, may be overly detailed for use by the pilot. This is because the pilot has much higher level thought than a computer system. A partial plan may also permit additional flexibility in a given situation by allowing the pilot to interpret the plan as necessary. For example, an instruction to increase airspeed may be accomplished by a negative change

in vertical velocity, by increasing the engine thrust, by changing the configuration, or any combination of the preceding methods.

Partial plans concentrate on a local strategy to revise faulty plans at the fail point. This means that the part of the plan that failed is analyzed to determine and correct the cause of failure and that, at least temporarily, the rest of the plan is preserved. This is usually more efficient than scraping an unsuccessful plan in its entirety. Furthermore, the reuse of old partial plans can be used by the pilot to evaluate the cause of the plan failure. The pilot or the plan generator could call for a change in global strategy when local strategy changes appeared inadequate.

Once a plan is generated, its execution could be simulated by using the cause-effect net. The simulation may expose a fault in the plan or an undesirable side-effect. If no unacceptable side-effects were detected and the plan had the intended effect, then the plan could be implemented by the pilot. A plan to restart an engine with a failed fuel boost pump could be tested by using the net in figure 2. The simulation would follow the flow of causality and determine if the desired goal was attained. If the goal was not attained, then the net would provide information about where the plan failed by the inconsistency across a node mentioned above. The simulation should increase the success rate as well as confidence in the plan.

4. THE EXECUTIVE

The primary function of the executive is priority resolution. The executive also controls the interface between the pilot and the computer. It processes the pilot's commands and directs the monitor and the plan generator, or accesses the knowledge base. During degraded mode operation, the executive insures that the monitor and the plan generator are operating on an accurate model of the aircraft. The intelligent executive should also have the capability to determine when a task is implicitly delegated to it by the pilot.

4.1 Priority Resolution

The problem is to take a massive amount of raw data and to process it in real time and produce relevant information in a form which is readily usable. Most information need not be presented to the pilot, unless specifically requested. Only essential information should be provided automatically. This will prevent an information glut.

A skilled pilot is able to stay ahead of his aircraft by rapid interpretation of his instruments. From practice, he knows where to look and also what to expect to see. While conventional instrumentation lends itself to this type of interpretation, multipurpose CRT displays do not. The pilot cannot look at a multipurpose display and know *a priori* what will be there. It could be engine data, navigation data, or

even emergency data. There are two distinct interpretations that must be made, first, what the information is and then second, what it means. If the handicap of the first interpretation is to be offset, then the quality of the information displayed must be very high. Clearly, there is more to the problem than format and organization factors. The next section is concerned with the content of the information provided. If the content is not relevant, then no amount of formatting or structure will increase the value of the information.

4.1.1 Priority by Content

The priority of any information from the monitor may be variable and depend upon the context of the current situation. The criticality of the loss of cabin pressurization depends upon the altitude of the aircraft. The loss of cabin pressurization at high altitude may be caused by loss of pneumatic pressure for air conditioning, cabin air outflow valve stuck open, or loss of structural integrity. It is clear that the loss of structural integrity of the cabin will require a descent. However, in the case of loss of pneumatic pressure, action other than a descent might restore cabin pressurization. If the loss of cabin pressurization is caused by a loss of pneumatic pressure due to an engine malfunction, then the cabin pressure might be most quickly recovered by correcting the engine problem. In practice, the pilot would probably initiate a descent while working on the engine. If

the same problem occurred at low altitude, no pilot action would be necessary and accordingly, the pressurization failure would not warrant a high priority. On the other hand, some warnings, such as an engine fire indication, should always have high priority. Table 1 in section 5.1 shows how scripts can be used to establish a tentative priority hierarchy.

The criticality of a generator malfunction depends upon the operation of the other generators and the engines. The failure of an engine results in the loss of its generator. The loss of the second generator is always more critical than the loss of the first. For most multi-engine aircraft, a single generator can supply all essential electrical power, but all non-essential electrical equipment must be shutdown to prevent overload and subsequent loss of the single generator. The same reasoning is likely to be true of any system which relies on redundancy for reliability.

An example of changing priorities can be seen during the normal takeoff sequence. Through the very early part of the takeoff roll, any deviation may be reported because the aircraft would be traveling slowly and would have sufficient runway to easily stop. However, during the critical phase of takeoff (from about 50 knots until after takeoff is completed), the executive should suppress any information that is not related to takeoff performance parameters, such as engine thrust, flight control and flight hydraulics, emergency electrical power, landing gear and brakes, and the inertial

reference system. The pilot should not be distracted with non-critical information during a critical phase of flight. The information could be provided after the takeoff is completed. Besides, the outright suppression of data, any data that is presented, is ordered in a priority ranking so that the pilot will receive the most critical information first. In the case of multiple emergencies, the system would be able to present up to four malfunctions or deviations at once. This would permit the pilot to change the order of priority as he might deem necessary.

Because all high workload or critical phases of flight are not always the same, the aircrew should receive an indication that a low priority deviation has been detected. The indication could be temporarily ignored with reasonable confidence, but if the pilot or another crew member could afford the time, the suppressed information could be called up for display.

4.1.2 Priority by Structure

The use of multi-purpose displays (MPD) has advantages of flexibility, free formatting, color, and small space requirements. The primary disadvantage is that it requires the pilot to make an additional interpretation. That is, the pilot must first determine what he is looking at before he can determine the meaning of the content. This is unlike a conventional system where the pilot knows that by looking at a

certain position, he will see a certain type of information.

As mentioned earlier, the content of the information is of primary importance. However, the organization of the material is also worth commenting upon. If we assume that the basic display layout to be somewhat as pictured in figure 5, the MPDs could be given an arbitrary ranking. For example, MPD-1 could be the display which always displays the information that is considered most important by the computer system. MPD-2 might have the next most important information. The pilot could easily determine his general situation by a very rapid and compact scan. Supplemental information could be provided on MPD-3 and MPD-4.

The Vertical Situation Display (VSD) should include not only attitude and pitot static information, but also engine thrust and projected flight path information. The Head-Up Display (HUD) should present the same information as the VSD. The primary difference would be the background against which it is presented. The HUD's background would be the natural visual scene out the front for the aircraft. The VSD's background would be computer generated, infra-red imaging, or radar profiles of terrain. The Horizontal Situation Display (HSD) would provide flexible information and formatting depending upon the context. Typical information would include terrain topography, airfield locations, significant traffic, and course information. Supplemental information could be made available depending upon the phase of flight. Other

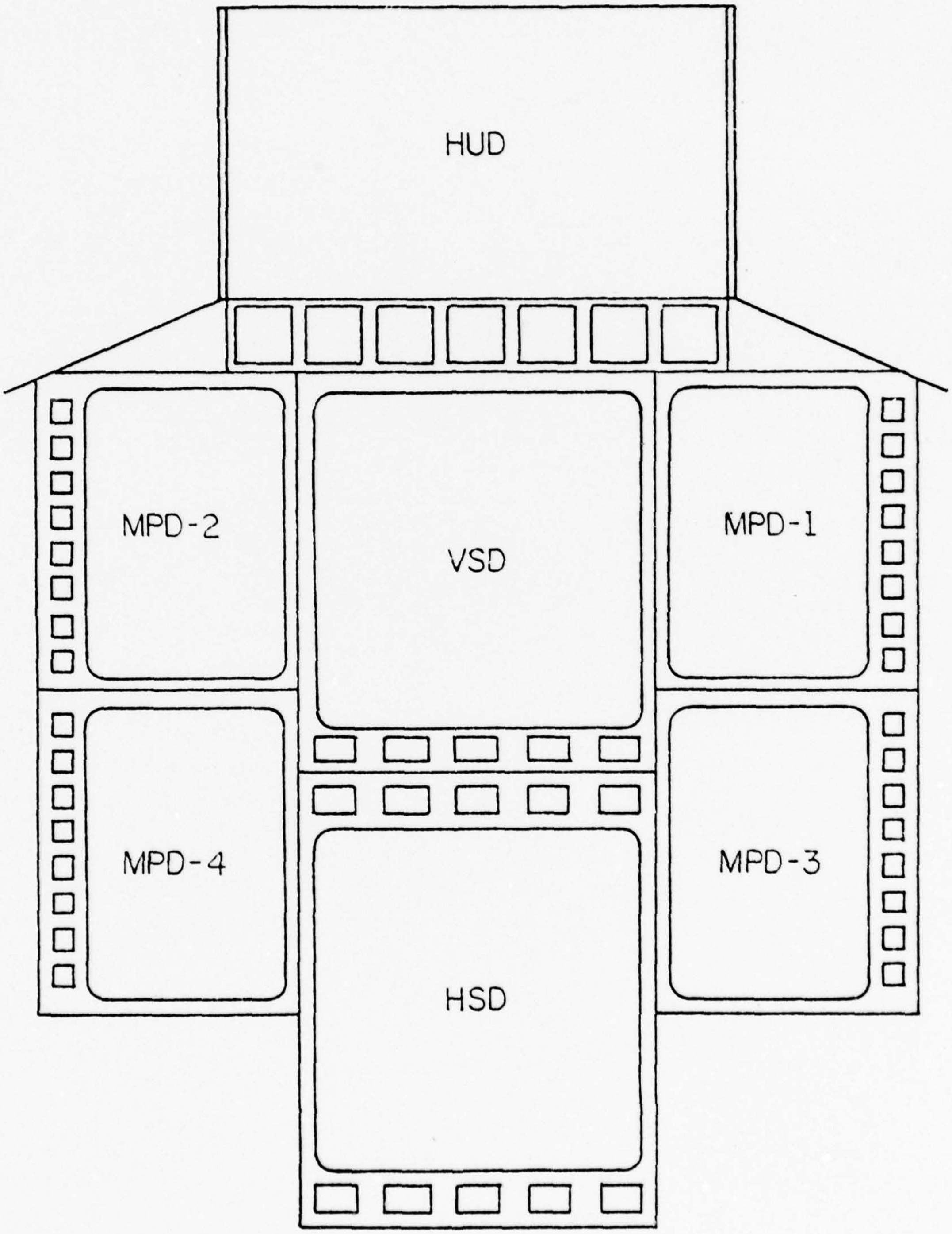


Figure 5 Front Panel of Cockpit

information which would be useful during takeoff would include an aircraft performance factor (scale of 0 to 1.0 relative to optimal acceleration), distance to liftoff, distance to stop (if takeoff is aborted), and runway remaining. Once safely airborne, the script would change and a new priority ranking would come into effect.

4.2 Degraded Mode Operation

The degraded mode operation is inherent with a condition sensitive monitoring system. Since the condition-based context will affect the priority and action, a separate model should not be necessary. A good example of degraded mode operation was given above for the failure of the second generator. It reflects the monitors sensitivity to the degraded mode model of the aircraft.

Besides changes in handling systems, other changes are probable in a degraded operation of the aircraft. Lower performance is common. The flight plan must be reviewed and necessary revisions made to reflect the degraded capabilities of the aircraft. If an emergency landing is anticipated the necessary preparations should be made. These might include a revised landing procedure, changing course to a closer airfield, conservation measures for fuel or other consumables, or relaying the necessary information to the radar controller for coordination.

4.3 Dynamic Task Allocation

The workload of the pilot is never constant and an intelligent onboard computer system should be able to assist without having to be asked. An example might be a descent to land with an intermediate altitude restriction because of high terrain. If the pilot did not begin his level off by the normal lead point, the computer system would respond by providing a warning. If the pilot did not respond to the warning, then the computer system would, after a reasonable interval, engage the autopilot and cause the plane to level off at the proper altitude. The pilot would have three distinct opportunities to prevent or override the computer system. First, he could have started the level off at or before the normal lead point and avoided the warning. Second, he could have responded to the warning by starting to level off, by specifying a change to the altitude clearance in the computer, or simply commanding the computer to "stand by". The last response by the pilot would prevent the computer from engaging the autopilot in an attempt to level off at the clearance altitude, but not from issuing a further warning message if necessary. Thirdly, the pilot could override the autopilot and resume normal control. This sequence demonstrates protection against both computer malfunction and pilot error.

Task allocation requires an extensive array of information in order to be effective. The information would include script and flight plan data on which to base the nominal conditions as well as the current flight clearance. The condition of the aircraft must be known to determine what limitations should be imposed upon the aircraft. For example, an aircraft with an engine out would have performance limitations such as turns into the failed engine, lower rate of climb and cruise altitude, and significantly reduced low speed performance.

5. THE KNOWLEDGE BASE

The basic functional entities share a common knowledge base. The main elements of the knowledge base will include

1. Scripts

2. Cause-effect nets

3. Real-time data
   a. Sensor data
   b. Data link

5.1 Scripts

Phase-based context information is characterized by its routine nature. This type of information can be provided by scripts. A script is a "structure that describes an appropriate sequence of events in a particular context" [16]. It is a mundane, stereotyped sequence for a well-known situation. It can provide default information which will expedite certain processes. In addition to information about the present context, it can provide information about a future context. For example, we normally anticipate that a descent will follow the cruise and that a descent will precede a landing. The following scripts provide a sample of the nature of a flight script. The high level script is quite vague. Like a tree structure, the lower levels become increasingly detailed.

The entry conditions are not mandatory preconditions which must be satisfied, but are guides which indicate when a script should be entered. The transitions indicates the

alternatives available. The transition script with the closest matching entry conditions would be the likely candidate for the following script. For example, if during script 4.2.3, glide slope, the selection of the next script would be either script 4.3, flare, or script 4.5, go-around/missed approach. The actual conditions of the aircraft would determine which script should be selected. The pilot would be able to override the computer selection of a script.

5.1.1 Top Level Flight Script

1. Preflight Operations

   1.1 Flight Planning
   1.2 Preflight Checkout
   1.3 Taxi

2. Takeoff

   2.1 Line Up
   2.2 Takeoff Roll
   2.3 After Airborne
   2.4 Departure
   2.5 Aborted Takeoff

3. Enroute

   3.1 Climb
   3.2 Cruise
   3.2 Descent

4. Landing

   4.1 Approach to Field
   4.2 Final Approach
   4.3 Flare
   4.4 Touchdown and Rollout
   4.5 Go-around/Missed Approach

5. Postflight Operations

   5.1 Taxi
   5.2 Postflight Checkout

5.1.2 Landing Script

4. Landing

4.1 Approach to field

Entry conditions: Script 3; At or near final approach fix altitude; Want to make an approach or landing at nearby airfield.

4.1.1 Calculate course to final approach fix.
Get approach procedures from flight plan.
Comply with current clearance.

4.1.2 Calculate optimal energy profile.
Compensate for winds.
Check for minimum terrain clearance.
Present information to pilot on HSD.

4.1.3 Monitor navigation to the final approach fix.

Normal transition: Script 4.2.
Alternate transitions: Script 4.5, Script 3.

4.2 Final Approach

Entry conditions: Script 4.1; Near the final approach fix.

4.2.1 Intercept the final approach course inbound.
Insure landing system radio signal is reliable.
Confirm landing clearance.

4.2.2 Glide slope
Intercept glide slope.
Establish landing configuration.
Maintain the glide slope and final approach course.
Maintain recommended approach angle of attack.

Normal transition: Script 4.3.
Alternate transition: Script 4.5.

4.3 Flare

Entry conditions: Script 4.2; Aircraft is at the flare point.

4.3.1 Decrease rate of descent.

4.3.1 Decrease airspeed.

4.3.3 Maintain runway alignment.

Normal transistion: Script 4.4.
Alternate transition: Script 4.5.

4.4 Touchdown and Rollout

Entry conditions: Script 4.3; Aircraft on the runway.

4.4.1 Check thrust at idle

4.4.2 Lower nosewheel to runway.

4.4.3 Begin braking as required.
Apply reverse thrust.
Extend spoilers.
Apply wheel brakes

4.4.4 Turn off runway.

Normal transition: Script 5.
Alternate transitions: Script 4.5, Script 2.2, Script 2.1..

4.5 Go-Around/Missed Approach

Entry conditions: Script 4; Excessive deviation of altitude, airspeed, course, or glide slope; Want to execute a missed approach.

4.5.1 Set throttle at takeoff power.

4.5.2 Configure for go-around.
Set flaps for go-around.
Raise landing gear (if aircraft is climbing).

4.5.3 Follow missed approach instructions.
Climb to missed approach altitude.
Maintain missed approach course.

Normal transition: Script 4.1.
Alternate transition: Script 2.

5.1.3 Priority Changes During Different Scripts

The script can provide information about priorities so that non-critical information may be temporarily suppressed. For example, during takeoff and landing, where excess thrust available is reduced and terrain clearance is minimal, any engine malfunction must be considered critical. Any delay in taking the proper action could result in the loss of the aircraft. A similar malfunction during cruise might not be as critical because of the safety margin provided by being at a high altitude. However, during high altitude cruise, a loss of cabin pressurization is very serious and requires immediate action. The same problem during takeoff would not be critical. The computer must be aware of changes in context to provide maximum assistance to the aircrew. The relative priority of certain situations may be determined in part by the phase-based context. The script can provide a priority structure for common situations in the form of a list. The basic areas of the list would be divided by the normal flight operations information. For example, a list of priorities for landing might be something like the following list. Note that during the cruise script, all deviations are of higher priority than normal operations, which has been set arbitrarily to a priority of four.

AREA OF MALFUNCTION	SCRIPTS					
	TAKEOFF ROLL		CRUISE		FINAL APPROACH	
	(VMC)	(IMC)	(VMC)	(IMC)	(VMC)	(IMC)
Flight controls	1	1	1	1	1	1
Engine	1	1	1	1	1	1
Double generator	2	1	2	1	2	1
Airspeed deviation	2	2	3	3	1	1
Low fuel, second warning	2	2	2	1	3	3
Wheel brakes	2	2	3	3	2	2
Single hydraulic	2	2	2	2	2	2
Single generator	3	3	3	3	3	3
Fuel boost pump	3	3	2	2	5	5
Normal operations	4	4	4	4	4	4
Cabin pressurization	6	6	1	1	7	7
Low fuel, first warning	5	5	3	3	7	7

NOTE: Priority 1 is the highest ranking and Priority 7 is the lowest.
VMC = Visual Meteorological Conditions
IMC = Instrument Meteorological Conditions

Table 1 Relative Priorities within Scripts

5.2 Cause-Effect Nets

The condition-based context reflects the state of the aircraft and its systems. Previous examples of the condition sensitivity were the failure of the second generator, the change in normal state during fuel transfer, and the engine flame-out caused by the malfunctioning fuel boost pump. Part of this information can be represented as a cause-effect net as depicted in figure 2, which utilizes a representation based upon the *Common-Sense Algorithm* [13, 14]. This net can provide cause-effect relations between system components and their respective states. It can also represent the relationship among high level systems. External or environmental context is characterized by its relative independence of the aircraft and phase of flight. The primary factor is the weather. The representation of the external context can also be represented in the form of a cause-effect net. This net should be independent of the previously discussed condition-based context relations because the external context is likely to change quite often.

5.3 Real-Time Data

5.3.1 Sensor Data

An extensive network of sensors would be needed to provide the monitor with the necessary data. The sensor network would probably use distributed processors to increase through-put. All of the sensors for a related system would be

channeled through a preliminary processing stage which would only forward data which was a change from the previous state or if a certain interval of time had passed without a report. This first stage processing would use a very low level model to verify the sensor output. After the first stage of processing, the data would go to a data acquisition unit which would receive verified sensor data from the first stage processors of related units. These data acquisition units would use a higher level model against which to verify sensor data. Here again, to reduce bus communications requirements, processed data would not be forwarded, unless a change were detected or a certain interval of time had passed without a report.

5.3.2 Data Link

The data link reduces the need for the pilot to feed information to the computer system, while providing real-time information to the computer. The data link will provide information that is not available from the onboard sensor system or from the onboard knowledge base. This would include flight clearance information for changes in flight level, speed, or routing. Another important type of information that would be available would be enroute and destination weather. Other information might include detailed terrain elevations, traffic, and alternate airfield data for the current flight sector.

During emergencies, the data link would reduce the communications requirements for the aircrew and permit them to concentrate on the actual problem because the computer system could use the data link to advise ground control facilities of the status of the aircraft. If necessary, the data link could be utilized to permit the ground-based computer to assume some of the processing functions in case of an onboard computer system malfunction.

6. CONCLUSION

The future use of airborne computer systems seems certain. Sufficient emphasis must be placed upon insuring that the design of future systems make maximum use of existing technology. The concepts developed in the area of artificial intelligence can have a profound impact on the capabilities of advanced computer systems. In this thesis, it has been argued that an intelligent airborne computer system can enhance flight safety by reducing the pilot's workload by

1. assuming the primary monitoring function and thereby relieving the pilot of this tedious, time-consuming, but simple task,
2. providing plans to assist the pilot cope with abnormal situations as well as normal operations,
3. prioritizing information so as to not inundate the pilot with information and especially irrelevant information.

The intelligence of the computer system is based upon its sensitivity to context. The context information in the knowledge base is based upon scripts and cause-effect nets. The scripts provide context for each particular phase of flight. The cause-effect nets provide context based upon the condition of the aircraft and its environment. The intelligent airborne computer system frees the pilot of trivial tasks to perform the high level management of the aircraft.

LIST OF REFERENCES

1. Buchanan, B. G., et al. "Heuristic Dendral: A Program for Generating Exploratory Hypotheses in Organic Chemistry", in Machine Intelligence 4, American Elsevier Pub. New York, 1969, pp. 209-254.

2. Boeing Company, Military Airplane Systems Division, Integrated Information Presentation and Control System Study, Volume I and II, Technical Report AFFDL-TR-70-79, August, 1970.

3. Chien, R. T., et al., Computer-Aided Decision-Making for Flight Operations, Technical Report Number 3, Report T-35, Coordinated Science Laboratory, University of Illinois, Urbana, Illinois, November, 1976.

4. Chien, R. T. and S. J. Weissman, "Planning and Execution in Incompletely Specified Environments", Proceedings for the Fourth International Joint Conference on Artificial Intelligence, Tblisi, USSR, September, 1975, pp. 89-127.

5. Davis, Paul R., Using and Re-using Partial Plans, Ph.D. Thesis, Department of Electrical Engineering, University of Illinois, Urbana, Illinois, June, 1977.

6. Della Salle, L., W. Butler, and T. Newman, "Aerospace Application of Microprocessors", NAECON '76 Record, 1976, pp. 307-313.

7. Frink, A. and R. Collender, "Future Trends in Aircraft Display/Control Systems", EASCON '76, 1976, pp. 113A-J

8. General Electric Co. Integrated Engine Instrument System, Final Technical Report, December 1973.

9. Hitt, E. F. and T. A. Brim, "Digital Avionics Information System Control and Displays Architecture", AIAA paper 77-1515, Proceedings of the AIAA Second Digital Avionics Systems Conference, Los Angeles, California, November, 1977.

10. Hughes Aircraft Company, Master Monitor Display Study, Final Report, Contract N62269-73-C-0138, Naval Air Development Center, January, 1974

11. McDonnell Douglas Corporation, DC-10 Flight Crew Operating Manual.

12. Moses, J. Symbolic Integration, MIT MAC-TR-47, December, 1976.

13. Reiger, C. The Commonsense Algorithm as a Basis for Computer Models of Human Memory, Inference, Belief and Contextual Language Comprehension, Technical Report 373, University of Maryland, College Park, Maryland, May, 1975

14. Reiger, C. "An Organization of Knowledge for Problem Solving and Language Comprehension", Artificial Intelligence, 7(1976), pp. 89-127.

15. Ropelewski, R. "Head-Up Display Systems Evaluated", Avaition Week and Space Technology, January 10, 1977, McGraw-Hill, Inc., New York, New York, pp. 70-79.

16. Schank, R. and R. Abelson, Scripts, Plans, Goals, and Understanding - An Inquiry into Human Knowledge Structures, Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1977

17. Shortliffe, E. H. Computer-Based Medical Consultation: MYCIN, American Elsevier Pub., New York, 1976.